

## CLAIM(S):

1. An architecture for developing a distributed information system, the architecture comprising:

a service definition tool for generating service protocols as a service definition, each service protocol including a plurality of messages, the messages including incoming messages and outgoing messages, each message carrying a plurality of data fields;

a component development tool for generating a first and a second plurality of components that implement and consume services, each component in the first plurality of components representing a physical entity in the distributed information system, each component in the second plurality of components representing a logical entity in the distributed information system;

a system development tool for generating a plurality of component instances based on the first and the second plurality of components; and

an engine software program for running on each of a plurality of networked nodes, the engine software program providing a programmable run-time environment for hosting the plurality of component instances and supporting communication between component instances.

2. The architecture of claim 1, wherein the services are defined by the service protocols, and wherein the component development tool allows implementations of the services in the first and the second plurality of components, the implementations of the services exposed as service provider ports and service consumer ports.

claims  
1-25  
OK for 7/7/00  
TS

005180" 1548E950

3. The architecture of claim 2, wherein service provider ports and service consumer ports based on the same service protocol are complimentary.
4. The architecture of claim 3, wherein the system development tool allows communication links to be defined between service provider ports and complimentary service consumer ports.
5. The architecture of claim 1, wherein at least one component instance is self-sufficient and functions without interacting with other component instances.
6. The architecture of claim 4, wherein a component instance includes at least one service provider port that allows multiple simultaneous communication links with complimentary service consumer ports.
7. The architecture of claim 4, wherein component instances are executed concurrently, and wherein the communications between service provider ports and complimentary service consumer ports are asynchronous.
8. The architecture of claim 4, wherein the communication links include one-to-one links, one-to-many links and many-to-many links, regardless of ports involved.
9. The architecture of claim 1, wherein the system development tool allows each component instance to be configured.
10. The architecture of claim 1, wherein the system development tool represents the distributed information system as a single entity, regardless of physical node and network composition into which the component instances will be deployed.

005130-1648960

11. The architecture of claim 10, wherein the system development tool deploys each component instance to one of the plurality of networked nodes.

12. The architecture of claim 11, and further comprising a local repository on each of the plurality of nodes, the local repository on each node storing data representing the component instances deployed to and hosted by that node and storing communication link data for the component instances deployed to and hosted by that node.

13. The architecture of claim 11, wherein the system development tool allows changes to be made to the component instances deployed to and hosted by the plurality of networked nodes and allows changes to be made to communication links between the component instances deployed to and hosted by the plurality of networked nodes.

14. The architecture of claim 13, wherein the system development tool allows deletion of the component instances deployed to and hosted by the plurality of networked nodes and allows deletion of communication links between the component instances deployed to and hosted by the plurality of networked nodes.

15. The architecture of claim 1, and further comprising a central system repository for storing the components, the component instances, link data, infrastructure configuration and configuration data for the service protocols.

16. The architecture of claim 1, wherein at least one of the component instances supports continuous activities internally.

17. The architecture of claim 1, wherein each of the component instances is configurable to participate in activities that are collectively performed by a plurality of component instances.

18. The architecture of claim 4, wherein the only run-time dependencies between component instances that communicate with each other are logical dependencies implemented using the component development tool.

19. A distributed information system comprising:  
a plurality of component instances connected by links; and  
a plurality of networked nodes running an engine software program, the engine software program providing a programmable run-time environment for hosting the plurality of component instances and supporting communication between component instances.

20. The distributed information system of claim 19, wherein the run-time environment acts as a container for the component instances, and wherein the run-time environment provides API implementations to the component instances.

21. The distributed information system of claim 19, and further comprising a local repository on each of the plurality of nodes, the local repository on each node storing component instances deployed to and hosted by that node and storing communication link data for component instances deployed to and hosted by that node.

22. The distributed information system of claim 21, wherein the local repository notifies component instances and links of configuration data changes.

23. The distributed information system of claim 20, wherein the run-time environment dynamically manages ports and links for the component instances and includes communication implementation for delivering messages between connected complimentary component instance ports.

24. The distributed information system of claim 19, wherein the engine software program supports communication with a system development tool that can be used to create, destroy and modify component instances and links.

25. The distributed information system of claim 21, wherein the programmable run-time environment dynamically changes according to configuration data stored in the local repository.

26. A method of modeling a distributed information system, the method comprising:

identifying physical and logical entities in the distributed information system;

organizing the identified entities into a hierarchical system model;

identifying services provided by the identified entities using the hierarchical system model;

creating service protocols, each service protocol defining one of the identified services;

implementing the identified services as a plurality of components using the service protocols, the plurality of components implemented as concurrent independent software entities, the implementation including generating service provider ports and service consumer ports to expose the implemented services;

0051801648950

populating the system model by creating instances of the implemented components;  
linking the component instances together; and  
assigning the component instances to a plurality of run-time software engines.

27. The method of claim 26, wherein the organization of the entities in the hierarchical system model is based on the generality and interdependence of the entities using a rule that assigns more generic entities lower in the hierarchical model than more specific entities.

005130-16482960